Annual Grant Progress Report
Award No NCC 2-1035:
Case-Based Capture and Reuse of Aerospace Design Rationale

Period covered by report: 5/1/98-9/20/30
Anticipated total funding period: 5/1/98–9/30/99

David B. Leake
Computer Science Department
Indiana University
Bloomington, Indiana, 47405
leake@cs.indiana.edu

The goal of this project is to apply artificial intelligence techniques to facilitate capture and reuse of aerospace design rationale. The project applies case-based reasoning (CBR) and concept mapping (CMAP) tools to the task of capturing, organizing, and interactively accessing experiences or "cases" encapsulating the methods and rationale underlying expert aerospace design. As stipulated in the award, Indiana University and Ames personnel are collaborating on performance of research and determining the direction of research, to assure that the project focuses on high-value tasks.

In the first five months of the project, we have made two visits to Ames Research Center to consult with our NASA collaborators, to learn about the advanced aerospace design tools being developed there, and to identify specific needs for intelligent design support. These meetings identified a number of task areas for applying CBR and concept mapping technology. We jointly selected a first task area to focus on: Acquiring the convergence criteria that experts use to guide the selection of useful data from a set of numerical simulations of high-lift systems.

During the first funding period, we developed two software systems. First, we have adapted a CBR system developed at Indiana University into a prototype case-based reasoning shell to capture and retrieve information about design experiences, with the sample task of capturing and reusing experts' intuitive criteria for determining convergence (work conducted at Indiana University). Second, we have also adapted and refined existing concept mapping tools that will be used to clarify and capture the rationale underlying those experiences, to facilitate understanding of the expert's reasoning and guide future reuse of captured information (work conducted at the University of West Florida).

The tools we have developed are designed to be the basis for a general framework for facilitating tasks within systems developed by the Advanced Design Technologies Testbed (ADTT) project at ARC. The tenets of our framework are (1) that the systems developed should leverage a designer's knowledge, rather than attempting to replace it; (2) that learning and user feedback must play a central role, so that the system can adapt to how it is used, and (3) that the learning and feedback processes must be as natural and as unobtrusive as possible. In the second funding period we will extend our current work, applying the tools to capturing higher-level design rationale.

1 Progress on the CBR Engine

The initial software system developed at Indiana University is a general-purpose case-based reasoning retrieval engine. The current version manipulates two types of cases. *Data cases* encapsulate information about specific simulation runs. The system can display the information in data cases in tabular or graphical form. *Strategy cases* encapsulate information about (1) the criteria—the informal "engineering metrics"—used by experts to determine convergence for practical problems, and (2) the design task for which they were selected. This reflects the fact that the choice of which engineering metrics to use depends on the task context. Storing information about the design task enables the user to inspect the link between convergence criteria and the circumstances under which they are selected, suggesting when particular criteria may be application in the future.

At present, tasks are described in textual form. In the next phase of the project, concept maps will be integrated with the CBR system and used as a medium for describing and browsing tasks, in order to provide a flexible way to organize information about rationale underlying the choice of a set of criteria for a given task.

In the current system, users can guide case retrieval by specifying desired individual features or desired convergence trends for particular parameters. Trend information from the retrieved cases is presented to the user in graphical form. The user may select a new set of convergence criteria either by (1) drawing a "convergence box" bounding the acceptable limits of variation, or by (2) specifying acceptable numerical limits, or by (3) having the system retrieve a strategy case and re-apply the criteria it encapsulates. As strategy cases are acquired, the capability to reuse prior strategies will enable novice users to benefit from the criteria used by prior experts to address similar tasks.

After retrieval, the user may edit cases, modifying or adapting features, and may enter feedback evaluating whether each retrieved case displayed the desired convergence behavior. In the current version of the system, this optional feedback is a combination of simple yes/no judgments and optional textual comments. The yes/no feedback provides a set of positive and negative examples to be used in the future as data for refining convergence criteria by standard machine-learning methods for concept learning.

A central emphasis of the current implementation effort is to provide a general foundation for broad applicability:

- The system is written in Java, to provide platform independence and compatibility with other AI tools (such as the CMAP software) and with existing design support programs being developed at Ames.

- The case base is represented in a database format which is readily adaptable to changes in information needs (e.g.,as a result of changing design approaches), or as changing analytic tools create new types of data to characterize a design. This format for the case-base will also facilitate the use of database systems to manage the case-base, enabling "scale-up" to large case bases.

- Both data cases and strategy cases are stored and manipulated by the same tools, demonstrating the generality of the methods to manipulate and learn cases containing multiple types of information.

Thus although the demonstrated version addresses one specialized problem, it brings to bear very general-purpose algorithms whose results depend solely on the variety of design data presented to them and the amount of informed feedback supplied to it by knowledgeable designers. For example, the system's flexibility will facilitate satisfying future needs such as capturing and reusing information about sequences of steps in the application of design tools, or the choice of particular components in an aircraft.

2 Progress on the Concept Map Editor

The software system being developed at the Institute for Human and Machine Cognition of the University of West Florida consists of a set of general tools to enable the construction of navigable models based on concept maps, over the Internet or locally on hard disk or CD-ROM. This system will support the description and examination of networks describing tasks and component arrangements. These networks will be used to organize data and strategy cases and will elucidate experts' conceptualizations of design tasks and strategies. The system is written in Java to be platform-independent and to facilitate its integration with the CBR system and NASA design tools.

More specifically, this effort has focused on:

- Development of a browser tool that will allow the user to navigate through concept map-based multimedia environments and case libraries. This browser will allow the user to transparently navigate through local or remote servers, accessing media anywhere on the Internet.

- Development of software tools necessary to run a server that will store concept maps and the corresponding media. Particularly relevant in the design of this software is the scalability of the server, allowing large number of connections accessing many models with extensive collections of media.

During the first five months of the project, most of the development effort has concentrated on finishing the major functionality of the "Core," a modular plug-in architecture.

The Core allows us to construct applications by first decomposing the application into individual pieces, implementing each piece as a module, and then constructing the application by assembling the required set of modules.

Completing the core now allows us to design the module that will interface with the case-based reasoning components. This module is expected to host the necessary libraries that will enable the case-based reasoning software to interact with the concept map application and vice-versa, enabling the capture and examination of rationale information in both forms.

The concept map application which will be used in the project is currently being rewritten as a set of modules that plug-in to the Core. The initial development of the application has been completed.

3 Plans for Second Funding Period

In the coming month we expect to make an initial integration of the CBR and concept mapping systems. In the final combined system, concept maps will be used to organize acquired cases in a form that can be browsed by other designers in order to leverage their own expertise by profiting from the stored prior experiences. When this integration has been completed, we, in conjunction with our ARC collaborators, will evaluate the capabilities of the combined system. In light of this evaluation we will then refine, extend and integrate the system with other advanced design tools developed at ARC.